On the exam you may use a ***definition/results sheet***. It may be several pages long if you wish, but it should be hand written or typed—no photocopies of stuff in the book. This is because creating the sheet will help you learn and retain the information better. It can have definitions and theorem on it, but cannot contain any worked examples.

These are the sorts of questions you can expect on Exam 2.

1. True/False questions about many of the topics.
2. Draw a Venn diagram showing the relationship between the following sets. (e.g. regular languages, languages generated by a CFG, etc.) You do not need to worry about where Turing-recognizable or Turing-decidable languages are in the diagram yet.
3. Let A = XXX
    a. Prove that A is (or is not) YYY.
    b. Give a ZZZ that WWW *A*.

    Where
    - XXX might be: *regular expression*, *set notation description*, *DFA*, *CFG*, etc.
    - YYY might be: *regular*, *context-free*, *decidable*, *Turing-recognizable*, etc.
    - ZZZ might be: *DFA*, *CFG*, *PDA*, *NFA*, *regular expression*, *Turing machine*, etc.
    - WWW might be: *generates*, *recognizes*, *decides*, etc.
4. Give an implementation-level description of a Turing machine that recognizes/decides the following language: (Insert description of language here).
5. Determine the simplest *computational model* that suffices for a given language B. Another way to think about it is to determine the smallest *class of languages* (among the ones we have discussed) that B belongs to. You will need to prove that it can be done (or does belong) with one model, but that it can't be done (or does not belong) with a simpler model.
6. Show that the collection of SSSS languages is closed under TTTTT, where
    - SSSSS might be: *Turing-recognizable*, *Turing-decidable*, *regular*, *context-free*, etc.
    - TTTTT might be: *union*, *intersection*, *complement*, *star*, *concatenation*, etc.